



Dollars

แคชเชียร์ต้องการทอนเงินให้ลูกค้า โดยที่แคชเชียร์มีเงินทอนเป็นเหรียญเท่านั้น และเพื่อให้ลูกค้าพอใจ แคชเชียร์จะต้องทอนเงินโดยใช้จำนวนเหรียญให้น้อยที่สุด เช่น ถ้าเหรียญที่มีเป็นเหรียญ 10, 5, 2 และเหรียญ 1 ดอลลาร์ โดยแต่ละเหรียญมีจำนวนไม่จำกัด กรณีแคชเชียร์ต้องการทอนเงิน 38 ดอลลาร์ แคชเชียร์จะต้องใช้เหรียญ 10 ดอลลาร์ จำนวน 3 เหรียญ เหรียญ 5 ดอลลาร์ จำนวน 1 เหรียญ เหรียญ 2 ดอลลาร์ จำนวน 1 เหรียญ และเหรียญ 1 ดอลลาร์ จำนวน 1 เหรียญ ดังนั้นแคชเชียร์ต้องทอนเงินโดยใช้เหรียญทั้งหมด 6 เหรียญ จงเขียนโปรแกรมเพื่อหาจำนวนเหรียญแต่ละเหรียญที่ต้องใช้ทอนเพื่อให้จำนวนเหรียญที่ใช้ทอนมีจำนวนน้อยที่สุด โดยเรียงตามชนิดเหรียญที่รับข้อมูลเข้ามา

ข้อมูลนำเข้า:

บรรทัดแรก เป็นจำนวนเต็ม 2 จำนวน คือ n ($1 \leq n \leq 1000$) ใช้กำหนดจำนวนเงินที่แคชเชียร์ต้องทอน และ k ($1 \leq k \leq 10$) แทนจำนวนชนิดเหรียญ

บรรทัดที่สองมีจำนวนเต็ม k จำนวน เป็นมูลค่าของเหรียญ

ข้อมูลนำออก:

จำนวน k บรรทัด แสดงจำนวนเหรียญในแต่ละชนิดเหรียญ โดยเรียงตามชนิดเหรียญที่รับเข้ามา

Input	Output
8 4 1 4 5 10	0 2 0 0
38 4 1 5 4 10	0 0 2 3
18 3 15 3 1	1 1 0

```
1  #include <stdio.h>
2  #define MAXTOTAL 10000
3  int nway[MAXTOTAL+1] = {0};
4  int detail[MAXTOTAL+1][10] = {0};
5
6  int main() {
7      int i, j, n, k, m, value, bestNumCoins, countCoins;
8      int coin[10];
9      int num[10], idx;
10     scanf("%d %d", &n, &k);
11     for (i=0; i<k; i++) scanf("%d", &coin[i]);
12     for (i=1; i<=n; i++) {
13         value = i;
14         bestNumCoins = MAXTOTAL+1;
15         for (j=0; j<k; j++) {
16             if (value - coin[j] >= 0) {
17                 countCoins = nway[value - coin[j]] + 1;
18                 if (countCoins < bestNumCoins) {
19                     bestNumCoins = countCoins;
20                     for (m=0; m<k; m++) num[m] = detail[value - coin[j]][m];
21                     num[j]++;
22                 }
23             }
24         }
25         nway[i] = bestNumCoins;
26         for (m=0; m<k; m++) detail[i][m] = num[m];
27     }
28     for (m=0; m<k; m++) printf("%d\n", detail[n][m]);
29     return 0;
30 }
31
```



Coins

แคชเชียร์ต้องการทอนเงินให้ลูกค้า โดยที่แคชเชียร์มีเงินทอนเป็นเหรียญเท่านั้น และเพื่อให้ลูกค้าพอใจ แคชเชียร์จะต้องทอนเงินโดยใช้จำนวนเหรียญให้น้อยที่สุด เช่น ถ้าเหรียญที่มีเป็นเหรียญ 10, 5, 2 และเหรียญ 1 บาท โดยแต่ละเหรียญมีจำนวนไม่จำกัด กรณีแคชเชียร์ต้องการทอนเงิน 38 บาท แคชเชียร์จะต้องใช้

เหรียญ 10 บาท จำนวน 3 เหรียญ

เหรียญ 5 บาท จำนวน 1 เหรียญ

เหรียญ 2 บาท จำนวน 1 เหรียญ

เหรียญ 1 บาท จำนวน 1 เหรียญ

ดังนั้นแคชเชียร์ต้องทอนเงินโดยใช้เหรียญทั้งหมด 6 เหรียญ จงเขียนโปรแกรมเพื่อหาจำนวนเหรียญที่น้อยที่สุดที่ใช้ในการทอนเงิน

ข้อมูลนำเข้า:

บรรทัดแรก เป็นจำนวนเต็ม 2 จำนวน คือ n ($1 \leq n \leq 10,000$) ใช้กำหนดจำนวนเงินที่แคชเชียร์ต้องทอน และ k ($1 \leq k \leq 10$) แทนจำนวนชนิดเหรียญ บรรทัดที่ 2 มีจำนวนเต็ม k จำนวน เป็นมูลค่าของแต่ละเหรียญที่มี ซึ่งจะ
เป็นตัวเลขจำนวนเต็มไม่เกิน 100

ข้อมูลนำออก:

จำนวนหนึ่งบรรทัดแสดงจำนวนเหรียญที่น้อยที่สุดในการทอนเงินของแคชเชียร์

Input	Output
8 4 1 4 5 10	2 (หมายเหตุ 4 บาท 2 เหรียญ)
38 4 1 5 4 10	5 (หมายเหตุ 10 บาท 3 เหรียญ, 4 บาท 2 เหรียญ)
18 3 15 3 1	2 (หมายเหตุ 15 บาท 1 เหรียญ, 3 บาท 1 เหรียญ)

```
1  #include <stdio.h>
2  #define MAXTOTAL 10000
3  int nway[MAXTOTAL+1] = {0};
4
5
6  int main() {
7      int i, j, n, k, value, bestNumCoins, countCoins;
8      int coin[10];
9      scanf("%d %d", &n, &k);
10     for (i=0; i<k; i++) scanf("%d", &coin[i]);
11     for (i=1; i<=n; i++) {
12         value = i;
13         bestNumCoins = MAXTOTAL+1;
14         for (j=0; j<k; j++) {
15             if (value - coin[j] >= 0) {
16                 countCoins = nway[value - coin[j]] + 1;
17                 if (countCoins < bestNumCoins) bestNumCoins = countCoins;
18             }
19         }
20         nway[i] = bestNumCoins;
21     }
22     printf("%d", nway[n]);
23     return 0;
24 }
25
```



Max Sequence

กำหนดให้ $\{a_1, a_2, \dots, a_n\}$ เป็นลำดับของจำนวนเต็ม และกำหนดให้ $\{a_i, a_{i+1}, \dots, a_j\}$ เป็นลำดับย่อยของลำดับดังกล่าวนี้ โดยที่ i และ j เป็นจำนวนเต็มบวก และ $1 \leq i \leq j \leq n$ หรือกล่าวอีกนัยหนึ่งคือ สมาชิกทุกตัวของลำดับย่อยต้องมีตำแหน่งต่อเนื่องกัน ลำดับย่อยอาจมีได้หลายชุด เมื่อหาค่าผลบวกของสมาชิกทุกตัวในลำดับย่อยแต่ละชุดผลบวกที่ได้อาจมีค่าแตกต่างกัน ลำดับย่อยที่มีผลบวกของสมาชิกสูงสุดเรียกว่า “ลำดับย่อยที่มีค่าสูงสุด” ซึ่งอาจมีเพียงชุดเดียวหรืออาจมีหลายชุดก็ได้ ในกรณีที่ลำดับย่อยที่มีค่าสูงสุดมีค่าน้อยกว่าหรือเท่ากับศูนย์ เรียกว่า “ลำดับย่อยว่าง (Empty sequence)”

ตัวอย่าง

ลำดับ $\{4, -6, 3, -2, 6, -4, -6, 6\}$ มีลำดับย่อยที่มีค่าสูงสุดเพียงชุดเดียว คือลำดับย่อย $\{3, -2, 6\}$ โดยผลบวกของลำดับย่อยมีค่าเท่ากับ 7

ลำดับ $\{-2, -3, -1\}$ ไม่มีลำดับย่อยใดที่มีผลบวกมากกว่าศูนย์ ถือว่ามีลำดับย่อยว่าง โจทย์

จงเขียนโปรแกรมเพื่อรับจำนวนของสมาชิกในลำดับและรับค่าสมาชิกทุกตัวของลำดับนั้น จากนั้นทำการคำนวณและแสดงผลเป็นลำดับย่อยที่มีค่าสูงสุด และผลบวกของสมาชิกของลำดับย่อยนั้นตามรูปแบบที่โจทย์กำหนด

ข้อมูลนำเข้า

บรรทัดแรกรับจำนวนเต็มบวก N ซึ่งเป็นจำนวนของสมาชิกในลำดับ โดยที่ $1 \leq N \leq 2500$

บรรทัดที่สอง รับค่าจำนวนเต็ม N ตัว, a_1, a_2, \dots, a_N โดยที่ค่า a_i คือค่าของสมาชิกลำดับที่ i ของลำดับนี้ ค่าของสมาชิกแต่ละตัวคั่นด้วยเครื่องหมายเว้นวรรคจำนวน 1 วรรค รับประกันว่า $-127 \leq a_i \leq +127$ สำหรับค่า a_i ใดๆ ในลำดับ

ข้อมูลส่งออก

ให้แสดงผลตามเงื่อนไขดังต่อไปนี้:

1. ในกรณีที่หาลำดับย่อยที่มีค่าสูงสุด ได้เพียงชุดเดียว ให้แสดงลำดับย่อยนั้น
2. ในกรณีที่หาลำดับย่อยที่มีค่าสูงสุดได้หลายชุด ให้แสดงเฉพาะชุดแรกที่พบเมื่อนับจากต้นลำดับ เช่นลำดับ $\{4, -6, 3, -2, 6, -4, -6, 6, -6, 4, -2, 5\}$ มีลำดับย่อยที่มีค่าสูงสุด 2 ชุด คือ $\{3, -2, 6\}$ และ $\{4, -2, 5\}$ ซึ่งมีค่าผลบวกของลำดับย่อยเป็น 7 เท่ากัน ในกรณีนี้ให้แสดงคำตอบเพียงคำตอบเดียว คือลำดับย่อยชุดแรกที่ พบ คือ $\{3, -2, 6\}$
3. การแสดงลำดับย่อยที่มีค่าสูงสุดให้แสดงสมาชิกของลำดับย่อยทั้งหมดในบรรทัดแรก โดยใช้เครื่องหมายเว้นวรรคคั่นระหว่างสมาชิกแต่ละตัวจำนวน 1 วรรค
4. บรรทัดที่สองให้แสดงผลเป็นผลบวกของลำดับย่อยที่มีค่าสูงสุดนั้น

5. ในกรณีที่ลำดับย่อยที่มีค่าสูงสุดเป็นลำดับย่อยว่าง ให้แสดงข้อความ "Empty sequence" โดยไม่ต้องแสดงลำดับย่อยและผลบวกของลำดับย่อยนั้น

Input	Output
8	3 -2 6
4 -6 3 -2 6 -4 -6 6	7
3 -2 -3 -1	Empty sequence

แหล่งที่มา การแข่งขันคอมพิวเตอร์โอลิมปิก สอวน. ครั้งที่ 2 มหาวิทยาลัยบูรพา



Search sum 1

การ์ตูนเรื่อง Hunter x Hunter เป็นการ์ตูนที่ออกวางขายมาแล้ว N เล่ม ทางร้านไม่ยอมขายการ์ตูนให้ ขาจร จึงได้ตั้งเงื่อนไขว่าถ้าต้องการซื้อเล่มที่ i จะต้องซื้อการ์ตูนตั้งแต่เล่มที่ 1 ถึงเล่มที่ i จะไม่มีการแบ่งขาย เป็นเล่มย่อยๆ คุณเดินเข้าร้านด้วยเงิน M บาท อยากรู้ว่าจะซื้อการ์ตูนไปอ่านมากที่สุดได้กี่เล่ม เกือบลืมบอกไป ว่าการ์ตูนแต่ละเล่มไม่จำเป็นต้องมีราคาเท่ากัน

ข้อมูลนำเข้า:

บรรทัดแรกมีจำนวนเต็ม N และ K แทนจำนวนการ์ตูน และจำนวนครั้งที่คุณเดินเข้าร้านการ์ตูน ($1 \leq N \leq 100,000$; $1 \leq K \leq 100,000$) จากนั้น N บรรทัดจะระบุราคาของหนังสือการ์ตูน กล่าวคือ ในบรรทัดที่ $1+i$ จะระบุจำนวนเต็มบวก C_i ($1 \leq C_i \leq 10,000$) แทนราคาของหนังสือการ์ตูนเล่มที่ i อีกแต่ละ K บรรทัดถัดไป ระบุจำนวนเต็มบวก แทนจำนวนเงินที่คุณมีในการเข้าร้าน กล่าวคือ ในบรรทัดที่ $1 + N + j$ จะระบุจำนวน M_j ($1 \leq M_j \leq 1,000,000,000$) แทนจำนวนเงินที่คุณมีในการเข้าร้านครั้งที่ j

ข้อมูลนำออก:

มีทั้งสิ้น K บรรทัด บรรทัดที่ j ระบุว่าถ้ามีเงิน M_j บาท จะซื้อหนังสือการ์ตูนได้กี่เล่ม

Input	Output
3 3	1
10	2
20	0
30	
15	
30	
7	

Search sum 2

3C
100

การ์ตูนเรื่อง Hunter x Hunter เป็นการ์ตูนที่ออกวางขายมาแล้ว N เล่ม ทางร้านไม่ยอมขายการ์ตูนให้ ขาจร จึงได้ตั้งเงื่อนไขว่าถ้าต้องการซื้อหนังสือเล่มติดกัน เช่น ซื้อเล่มที่ 10 ถึงเล่มที่ 30 เป็นต้น กล่าวโดยละเอียดคือ ในการซื้อการ์ตูนจะต้องระบุจำนวนเต็ม i และ j ที่ $1 \leq i \leq j \leq N$ ในการซื้อดังกล่าวจะได้การ์ตูนเล่มที่ i ถึงเล่มที่ j คุณเดินเข้าร้านด้วยเงิน M บาท อยากรู้ว่าจะซื้อการ์ตูนไปอ่านมากที่สุดได้กี่เล่ม เกือบลืมบอกไปว่า การ์ตูนแต่ละเล่มไม่จำเป็นต้องมีราคาเท่ากัน

ข้อมูลนำเข้า:

บรรทัดแรกมีจำนวนเต็ม N และ K แทนจำนวนการ์ตูน และจำนวนครั้งที่คุณเดินเข้าร้านการ์ตูน ($1 \leq N \leq 100,000$; $1 \leq K \leq 100,000$) จากนั้น N บรรทัดจะระบุราคาของหนังสือการ์ตูน กล่าวคือ ในบรรทัดที่ $1+i$ จะระบุจำนวนเต็มบวก C_i ($1 \leq C_i \leq 10,000$) แทนราคาของหนังสือการ์ตูนเล่มที่ i อีกแต่ละ K บรรทัดถัดไประบุจำนวนเต็มบวก แทนจำนวนเงินที่คุณมีในการเข้าร้าน กล่าวคือ ในบรรทัดที่ $1 + N + j$ จะระบุจำนวน M_j ($1 \leq M_j \leq 1,000,000,000$) แทนจำนวนเงินที่คุณมีในการเข้าร้านครั้งที่ j

ข้อมูลนำออก:

มีทั้งสิ้น K บรรทัด บรรทัดที่ j ระบุว่าถ้ามีเงิน M_j บาท จะซื้อหนังสือการ์ตูนได้กี่เล่ม

Input	Output
4 4	3
17	2
10	2
20	0
30	
50	
30	
29	
7	

ลำดับเพิ่มขึ้นที่ยาวที่สุด 1 (Longest1)

ให้ลำดับ S ที่มีสมาชิก N จำนวน คือ x_1, x_2, \dots, x_N ให้หาลำดับย่อยของ S ที่เป็นลำดับเพิ่มขึ้นที่ยาวที่สุด โดยที่ลำดับเพิ่มขึ้น คือ ลำดับของจำนวน a_1, a_2, \dots, a_k ที่ $a_i < a_j$ สำหรับทุกดัชนี i และ j ที่ $i < j$

ข้อมูลนำเข้า:

บรรทัดแรกระบุจำนวนเต็ม N ($1 \leq N \leq 1,000$) จากนั้นอีก N บรรทัดระบุลำดับ S กล่าวคือบรรทัดที่ $1 + i$ สำหรับ $1 \leq i \leq N$, จะระบุจำนวนเต็ม x_i ที่ $(-1,000,000 \leq x_i \leq 1,000,000)$

ข้อมูลนำออก:

มีบรรทัดเดียวเป็นความยาวของลำดับเพิ่มขึ้นที่ยาวที่สุด

Input	Output
5 1 2 2 3 4	4
5 1 -1 2 -2 3	3

ลำดับเพิ่มขึ้นที่ยาวที่สุด 2 (Longest2)

ให้ลำดับ S ที่มีสมาชิก N จำนวน คือ x_1, x_2, \dots, x_N ให้หาลำดับย่อยของ S ที่เป็นลำดับเพิ่มขึ้นที่ยาวที่สุด โดยที่ลำดับเพิ่มขึ้น คือ ลำดับของจำนวน a_1, a_2, \dots, a_k ที่ $a_i < a_j$ สำหรับทุกดัชนี i และ j ที่ $i < j$

ข้อมูลนำเข้า:

บรรทัดแรกระบุจำนวนเต็ม N ($1 \leq N \leq 1,000$) จากนั้นอีก N บรรทัดระบุลำดับ S กล่าวคือ บรรทัดที่ $1 + i$ สำหรับ $1 \leq i \leq N$, จะระบุจำนวนเต็ม x_i ($-1,000,000 \leq x_i \leq 1,000,000$)

ข้อมูลนำออก:

มีสองบรรทัด บรรทัดแรกเป็นความยาวของลำดับเพิ่มขึ้นที่ยาวที่สุด ในบรรทัดที่สองให้ตอบลำดับเพิ่มขึ้นที่ยาวที่สุดดังกล่าว โดยตอบจำนวนเต็มในลำดับนั้น ไล่ไปตามลำดับ คั่นระหว่างจำนวนด้วยช่องว่างหนึ่งช่อง ถ้ามีหลายคำตอบ ให้แสดงคำตอบที่ปรากฏขึ้นก่อนในลำดับตัวเลข

Input	Output
5 1 2 2 3 4	4 1 2 3 4
5 1 -1 2 -2 3	3 1 2 3



Jump game

กบตัวหนึ่งถูกเลี้ยงในกล่องที่มีช่องย่อย ๆ ติดต่อกันเป็นแนวยาวโดยแต่ละช่องจะมีจำนวนเม็ดอาหารไม่เท่ากัน ถ้ากบกินอาหาร n เม็ดจะทำให้กบกระโดดได้ n ช่อง ซึ่งกบจะเลือกกินอาหารอย่างน้อย 1 เม็ด แต่ไม่เกินจำนวนอาหารที่มีในช่องนั้น อยากทราบว่ากบต้องกินอาหารน้อยที่สุดกี่เม็ดจึงจะสามารถกระโดดไปยังช่องสุดท้ายของกล่องได้ (เริ่มต้นกบอยู่ในช่องแรกของกล่อง และการันตีว่ากบจะสามารถกระโดดไปยังช่องสุดท้ายได้เสมอ)

ข้อมูลนำเข้า:

บรรทัดแรกระบุจำนวนช่อง N ($1 \leq N \leq 1,000$) จากนั้นอีก N บรรทัดระบุจำนวนอาหารในแต่ละช่อง กล่าวคือ บรรทัดที่ $1 + i$ สำหรับ $1 \leq i \leq N$, จะระบุจำนวนเต็ม x_i ($0 \leq x_i \leq 1,000,000$)

ข้อมูลนำออก:

มีหนึ่งบรรทัด แสดงจำนวนเม็ดอาหารที่น้อยที่สุดที่กบกิน

Input	Output
5 2 3 1 1 4	2
5 4 3 2 1 4	1



รหัสพันธุกรรม (Genetic code)

ห้องปฏิบัติการชีวภาพแห่งหนึ่งได้ทำการจัดเก็บรหัสพันธุกรรมของมนุษย์ต่างดาวที่มีการค้นพบในแถบเทือกเขา แต่ด้วยความยากลำบากของการสกัดพันธุกรรมทำให้รหัสพันธุกรรมที่สกัดได้มีการชำรุดเสียหาย โดยรหัสพันธุกรรมจะประกอบด้วยตัวเลข 0 และ 9 ยาวต่อเนื่องกัน และเพื่อความก้าวหน้าทางวิทยาศาสตร์ในอนาคต นักวิทยาศาสตร์ได้ทำการทดสอบการเข้ากันได้ของรหัสพันธุกรรม โดยนักวิทยาศาสตร์ต้องการทราบว่ารหัสพันธุกรรมสองรหัสมีความสัมพันธ์กันหรือไม่

ข้อมูลนำเข้า:

บรรทัดแรกจะเป็นจำนวนรหัสพันธุกรรม 1 ความยาวไม่เกิน 100 หลัก

บรรทัดที่สองจะเป็นจำนวนรหัสพันธุกรรม 2 ความยาวไม่เกิน 100 หลัก

ข้อมูลนำออก:

จำนวนตำแหน่งที่รหัสพันธุกรรมทั้งสองเหมือนกัน

Input	Output
12345612 23489	3 คือ 234
67812123 81234	4 คือ 8123